

Enhancement of Secrecy of Block Ciphered Systems by Deliberate Noise

Yahya S. Khiabani, Shuangqing Wei, Jian Yuan and Jian Wang

Abstract—This paper considers the problem of end-end security enhancement by resorting to deliberate noise injected in ciphertexts. The main goal is to generate a degraded wiretap channel in application layer over which Wyner-type secrecy encoding is invoked to deliver additional secure information. More specifically, we study secrecy enhancement of DES block cipher working in cipher feedback model (CFB) when adjustable and intentional noise is introduced into encrypted data in application layer. A verification strategy in exhaustive search step of linear attack is designed to allow Eve to mount a successful attack in the noisy environment. Thus, a controllable wiretap channel is created over multiple frames by taking advantage of errors in Eve's cryptanalysis, whose secrecy capacity is found for the case of known channel states at receivers. As a result, additional secure information can be delivered by performing Wyner type secrecy encoding over super-frames ahead of encryption, namely, our proposed secrecy encoding-then-encryption scheme. These secrecy bits could be taken as symmetric keys for upcoming frames. Numerical results indicate that a sufficiently large secrecy rate can be achieved by selective noise addition.

Index Terms—DES cipher, CFB mode, deliberate noise, linear cryptanalysis, Markov chain, wiretap channel, secrecy capacity.

I. INTRODUCTION

Traditionally, end-end secrecy delivery relies on symmetric or asymmetric encryption residing in the upper layer of a communication system, as well as sophisticated key management schemes [1], [2]. Without requiring a secure cipher, Wyner-type secrecy encoding provides a completely different solution to link-wise secret message delivery by random binning tailored to some presumed wiretap channel models in physical layer [3], [4]. In this paper, we propose an encoding-encryption approach to end-end secrecy delivery by encoding over a degraded wiretap channel across super-frames transmitted in the application layer. The resulting wiretap channel is created by injecting controllable noise into ciphertext after encryption, and determined by both the adversary node's uncertainty about the key of cipher and its limited resources in launching cryptanalysis. Secrete information transmitted in such manner could be taken as keys for the subsequent super-frame.

In the proposed framework, we are essentially exploring the techniques developed for physical layer secrecy encoding and cryptanalysis against symmetric block ciphers to serve our purpose of realizing end-end secrecy enhancement without

resorting to exogenous physical channel conditions. More specifically, Data Encryption Standard (DES) block cipher working in Cipher Feedback Mode (CFB) is taken to encrypt messages encoded using the Wyner type secrecy encoding scheme and then transmitted over multiple frames encrypted using different keys. Random binary noise is then deliberately added onto ciphertext, which are received by both legitimate user and an eavesdropper without any additional distortion. Such a hierarchical encoding-encryption framework allows us to transmit secrete messages over the resulting degraded wiretap channels in the application layer without making any assumption regarding end-end physical channel conditions.

In order to analyze secrecy enhancement achieved by utilizing our encoding-then-encryption approach, we need to study how Eve responds to the existing noise in her gathered data, and how that influences her cryptanalysis performance. In our case, Eve attempts to mount her linear attack with accumulated noisy ciphertexts, and thus applies a new verification strategy in the second phase of the linear attack while considering her possible resource constraints. Our statistical analysis shows that even when she uses a numerically optimized attacking strategy to obtain the key, it is likely for her to make mistakes in cryptanalysis. These possible failures of Eve over multiple frames make her channel degraded than the main channel, which can be further exploited by secrecy encoder to send additional secret bits over a super-frame. Therefore we could utilize generated secret bits over the last super-frame, whose secrecy is ensured by Wyner-type secrecy encoding scheme, to establish keys for next coming frames. The secrecy capacity of the system is computed assuming known channel states at Bob and Eve. Numerical results illustrate how deliberately added noise influences secrecy rate which can be further maximized at certain noise rate. It should be noted that the primary goal of our paper is to demonstrate through such a case-study how secrecy encoding and symmetric encryption could be put together to enhance end-end security, and thus we only provide capacity computation of the resulting channel towards the end without dealing with the implementation of a particular secrecy encoder [5].

In literature, very few analytical approaches have focused on the impact of noisy ciphertexts on the attacking performance. In [6] different security schemes are analyzed from both reliability and secrecy perspectives in the presence of channel noise; nonetheless, they do not discuss what modified strategy Eve needs to take adaptively against degradation, and nor have they considered further leveraging adversary's failures in its cryptanalysis. In fact, our approach shares a common spirit with friendly jamming schemes proposed in physical

¹Y. Khiabani and S. Wei are with the Department of Electrical and Computer Engineering, Louisiana State University, Baton Rouge, LA 70803, USA (Email: ysowti1@tigers.lsu.edu; swei@lsu.edu). Their work is supported in part by the Board of Regents of Louisiana under contract LEQSF(2009-11)-RD-B-03. J. Yuan and J. Wang are with the Department of Electronic Engineering, Tsinghua University, Beijing, P. R. China, 100084. (E-mail: jyuan, jian-wang@tsinghua.edu.cn)

layer secrecy encoding [7], [8] where deliberate noise is introduced in physical layer to interfere both legitimate link and eavesdropped link to improve the secrecy rate region. Unlike these works where link-wise physical channel features are explored to create a degraded wiretap channel, we essentially explore the adversary's disadvantages due to its uncertainty about the secret key bits and resulting deteriorated success rate in cryptanalysis in the presence of deliberate noise.

In addition, deliberate additive noise in encryption process was used to improve security of ciphers in previous works [9], [10], [11]. The primary goals in these works were to enhance the secrecy of a cipher by random binning and additive noise, not the one we are interested in, namely, deploying encoding-then-encryption framework to enhance secrecy by further encoding over a resulting degraded wiretap channel. Random measurement noise has also been considered in side channel attacks (SCA) where information about cryptographic operation is leaked through some physical measurements conducted by an adversary [12]. In [13], authors proposed to use multi-linear approximation utilized in Differential Power Analysis (DPA)-like attacks, which is powerful due its robustness against noise, to attack a symmetric cipher hardware by power analysis

The paper is organized as follows. In sections II, a preliminary description of CFB mode and linear cryptanalysis is provided. In section III the proposed security scheme is described in detail, and in IV, we design an optimized verification strategy for Eve. In section V the main channel and wire-tap channel are modeled and then the secrecy capacity of the resulting degraded wiretap channel is found in section VI. Finally, we present the numerical results in section VII and conclude the work in section VIII.

II. REVIEW OF RELEVANT BACKGROUND

A. Properties of CFB Mode of Operation with DES Cipher

DES is a symmetric key encryption cipher which has plaintexts and ciphertext of size 64-bit with the key length of 56 bit. Although DES is replaced by AES in some applications, it is still used and studied in many networks [14] and [15]. CFB mode is one of the operational modes that can be used to derive a key stream from block ciphers like DES [16]. We assume that block size in CFB mode is 64-bit. As can be seen in Fig. 1, at time n , encryption of previous ciphertext block C_{n-1} generates the key stream S_n which Xors with the current 64-bit plaintext P_n , to generate 64-bit ciphertext block C_n , i.e. $C_n = S_n \oplus P_n$ where $S_n = E_K(C_{n-1})$.

DES encryption is very sensitive to the noise introduced into ciphertexts or key bits. In particular, when one bit of the key or the input to the cipher is altered, it can deteriorate about half of the cipher output. This property is called avalanche effect [17]. However, since S-boxes in DES cipher are not ideal, the resulted bit error rate by avalanche effect is not exactly 0.5. This is also true in more advanced ciphers like AES as discussed in [18]. That is why in our analysis we assume that when there is an error in cipher input or in the key, each output bit is flipped with the probability of α .

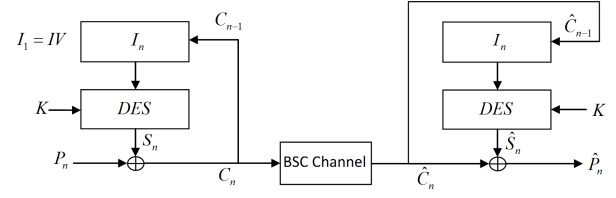


Fig. 1. Cipher feedback mode (CFB) with DES cipher [6]

B. Linear Cryptanalysis

Linear cryptanalysis is a known plaintext attack which was first proposed by Matsui in [19] to attack DES. It is one of the most widely used attacks on block ciphers. This cryptanalysis approach exploits a linear equation with the probability of $p \neq \frac{1}{2}$ which involves some input and output bits of the DES cipher and some key bits. The quantity $\varepsilon = |p - \frac{1}{2}|$, which is called bias, measures the correlation among plaintext, ciphertext and the key bits, and can be used as a criterion to distinguish the right key. Before attack, Eve has to gather a large number of plaintext/ciphertext pairs, and then for each possible key value compute its corresponding bias by counting the number of pairs that satisfy the linear equation.

If we refer to m as the number of attacked key bits in linear cryptanalysis, the number of subkey candidates would be 2^m that need to be sorted from rank 1 to 2^m based on their corresponding probability biases. It should be noted that it is not necessarily always true that the right key ranks the highest, but it will be surely among high ranked candidates. Assume that adversary only checks top 2^{m-a} candidates during exhaustive search, and since each subkey candidate gets checked with all possible combinations of $56 - m$ remaining unattacked bits, Eve has to run exhaustive search with at most 2^{56-m} encryptions for each candidate. As a result, the total number of 56 key bits examined in linear attack with bit advantage a is 2^{56-a} . In [20], A. Selçuk showed that when the total number of gathered plaintext/ciphertext pairs N are large enough, the probability of success P_s , defined as the probability that the right key is among 2^{56-a} top candidates, can be derived as

$$P_s = \Phi(2\sqrt{N}\varepsilon - \Phi^{-1}(1 - 2^{-a-1})), \quad (1)$$

where a is the bit advantage of the attack, ε is the bias of the used linear approximation and Φ is the cumulative distribution function of the standardized normal distribution.

III. THE PROPOSED SCHEME FOR SECURITY SYSTEM

Fig. 2 illustrates the proposed scheme for secrecy improvement in which after encryption of the original message S , intentional noise is injected into it to generate a degraded wiretap channel. Since we consider end to end secrecy, physical channel is assumed to be error-free. Therefore, the ciphertexts that Bob obtains only include errors caused by intentional noise introduced into encrypted data in application layer with bit error rate of η . Moreover, because Alice and Bob agree on the key used for the current data frame, Bob can decrypt the obtained noisy ciphertexts and then apply the wiretap

channel decoding algorithm that allows him to recover the original message \hat{S} with arbitrarily small error probability. As indicated in Fig. 2, there exists an oracle which is located after encryption and noise injection, whereby Eve can query and obtain consecutive plaintext/ciphertext pairs. However, due to the deliberate noise, the virtual oracle provides Eve with noisy ciphertexts distorted by a binary noise sequence with independent errors of rate η . The main advantage that Bob has over Eve is that Bob and Alice share the same encryption and decryption key which is unknown to Eve. Therefore, Eve has to adopt an attack strategy that can exploit the gathered noisy data in order to guess the secret key.

We assume that legitimate users initialize with a shared set of keys in a highly secure manner at the beginning. As a result, Alice can divide the whole data into equal size data frames, each including M number of data blocks of size 64-bit which is the block size used in CFB mode. In this way, the same key will be used for M 64-bit blocks in each frame for encryption and decryption at the receiver end. In this paper, we show that due to Eve's resource constraints, it is likely for her to make mistakes in assessing a frame key. As a result, Eve's channel is a degraded version of the main channel. We can leverage this advantage by applying Wyner secrecy encoding over super-frames to average over all possible failures by Eve. In Wyner-type encoder redundancy is added to correct errors that occur across the main channel, and randomness is added for keeping Eve ignorant across the wiretap channel [3], [5]. Note that this scheme can be generalized for other block ciphers like AES when they are used in operational modes like CFB or CBC (Cipher Block chaining).

Another issue is key scheduling problem to provide highly confidential and distinctive keys for each frame while Bob is fully aware of them. Here, we can use traditional way of key management which is sophisticated and costly. For instance, master/session key scheduling approach which is proposed for DES cipher in [1], [2]. In this technique, there exists a master key out of which frame keys as session keys can be originated. In our scheme, we propose a simpler approach which requires less expenditure. In this technique the secrecy required for frame keys is originated from secret bits delivered by Wyner secrecy encoder over the intentionally created wiretap channel. As a result, since encoder is performed over each super-frame, Alice can use input to the encoder to derive frame keys in next super-frame, for instance by applying a universal class of Hash functions [21], where the utilized function for each frame is publicly known. Bob is able to decode encrypted data and obtain the encoded message, and thus he will be able to derive keys for next frames. Note that the requirement for this approach is that there has to exist some root keys to initiate the keys for the first super-frame.

IV. EVE'S ATTACK STRATEGY AND ITS ANALYSIS IN NOISY ENVIRONMENT

This section studies the effect of the channel degradation on the performance of the linear cryptanalysis in terms of Eve's success rate. Since linear cryptanalysis is a known plaintext attack, Eve has to rely on the received plaintext/ciphertexts

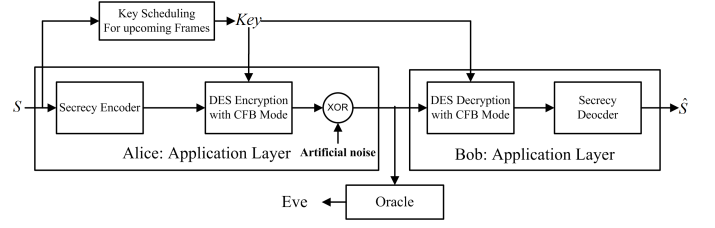


Fig. 2. The proposed security scheme based on the intentional noise

pairs. Due to the existing errors in these ciphertexts, when Eve examines a key, she is unable to distinguish between errors caused by the received noisy ciphertext and the ones induced by using the wrong key. Thus, she needs to design a new verification approach whereby she can find the right key. It should be noted that this attack strategy with verification process has to be designed in a way that attack success rate gets maximized from Eve's perspective.

A. Designed Verification Strategy for Attack

Consider ciphertexts go through a binary symmetric channel whose cross-over probability is η . As seen in Fig. 1, after C_n passes through channel, and Xors with channel noise, the received noisy 64-bit ciphertext \hat{C}_n will have error with the probability of $1 - (1 - \eta)^{64}$. Therefore, Eve can not rely only on two successive ciphertexts to check the correctness of a key, because they might have errors that can lead her to make mistakes. Indeed, Eve has to try a number of successive pairs, using CFB mode in order to increase her success rate.

In Fig. 3, two consecutive stages of CFB that are used to check the key are shown, where P_i and \hat{C}_i are respectively the plaintext and ciphertext for the i^{th} stage, S_i is the encrypted result of \hat{C}_{i-1} that after Xor with P_i generates \hat{C}_i^h . Provided that the used key is correct, \hat{C}_i^h must be the same as \hat{C}_i . However, due to the possible errors in \hat{C}_i or \hat{C}_{i-1} there might be some differences between \hat{C}_i and \hat{C}_i^h even though the used key is right. Therefore, Hamming Weight (HW) of Xor of \hat{C}_i^h and the ciphertext \hat{C}_i denoted by E_i must be compared with a threshold denoted as τ . Then, a key trial for the i^{th} stage can be considered successful if this HW is less than τ .

Note that at stage i when there is an error either in the input to the cipher, i.e. \hat{C}_{i-1} or in the key, there will be burst of errors in S_i , which makes \hat{C}_i^h totally different and in special case of $\alpha = 0.5$ independent from \hat{C}_i . Therefore, by choosing a small value for threshold τ and comparing HW of $\hat{C}_i^h \oplus \hat{C}_i$, Eve can know that either input to the cipher or the key is noisy. In Table I, the key verification strategy for Eve is given that she needs to follow in the brute-force attack phase of linear cryptanalysis to test the correctness of the examined key k_i . In this strategy, Eve examines each key candidate N_c times with N_c consecutive pairs. When at least one of trials is successful, Eve decides that the key is correct. That is because for a correct key, N_c is chosen such that she can make sure that with a high probability at least in one trial out of N_c tests, input to the cipher has no error that results in a success.

Now the question is how we can choose the optimum value for τ . When the tested key is right, at stage i , for error-free

TABLE I
VERIFICATION STRATEGY

- 1- Pick N_c number of consecutive pairs.
- 2- Try N_c chosen pairs over N_c chained CFB stages using the key k_i .
- 3- A trial is successful if $HW(E_i = \hat{C}_i \oplus \hat{C}_i^h) \leq \tau$.
- 4- If there exists at least one successful event out of N_c trials, k_i is the correct key, otherwise it is wrong.

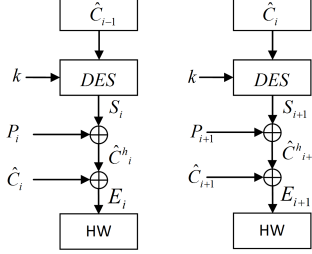


Fig. 3. key verification process for Eve with two consecutive CFB stages

\hat{C}_{i-1} , \hat{S}_i will be error free and all the errors in \hat{C}_i^h will be caused by the possible errors in \hat{C}_i . However, we can choose τ such that with a high probability, the number of errors in \hat{C}_i does not exceed this threshold. Hence, the minimum possible value for τ has to be adjusted such that at stage i , the probability that the number of bit errors in \hat{C}_i exceeds τ becomes negligible. This probability is denoted by P_{fault}

$$P_{fault} = 1 - \sum_{i=0}^{\tau} \binom{64}{i} (\eta)^i (1 - \eta)^{64-i}. \quad (2)$$

In the next step, we need to find the optimum value for N_c . Suppose that Eve tries a key to see if it is the right one, and let K_0^h be the hypothesis when the key is wrong and K_1^h when it is right. Then, we introduce random variable A_i such that $A_i = 1$ defines successful trial at the i^{th} stage that happens when Hamming weight of E_i is less or equal to τ , and $A_i = 0$ otherwise. By proper selection of τ , We can make sure that whenever there is no error in the input to the cipher, Eve can recognize the right key. Hence, the probability of having a successful event at the i^{th} stage given the right key will be

$$P_1 = Pr[A_i = 1 | K_1^h] = P[\hat{C}_{i-1} \text{ is error-free}] = (1 - \eta)^{64}. \quad (3)$$

All N_c tests will fail if in all of these trials, inputs to the ciphers have error. If it happens when the key is right, Eve will miss it, which has the probability of

$$P_m = (1 - P_1)^{N_c}, \quad (4)$$

We call P_m key missing probability. Thus, we need to find minimum N_c such that keeps P_m below a threshold like T_m .

Now we need to compute the probability that Eve mistakenly admits a wrong key while examining a single candidate. When the used key is wrong due to the avalanche effect, \hat{C}_i^h will have bit error rate of α , that after Xor with \hat{C}_i with bit error probability of η , results in output bit error rate of γ as

$$\gamma = \alpha(1 - \eta) + \eta(1 - \alpha). \quad (5)$$

Since to admit a wrong key at the i^{th} stage as the right one, HW of E_i must be less than τ , the probability of a successful

trial at this stage for a wrong key is

$$P_2 = Pr[A_i = 1 | K_0^h] = \sum_{i=0}^{\tau} \binom{64}{i} \gamma^i (1 - \gamma)^{64-i}. \quad (6)$$

On the other hand, Eve accepts a wrong key when there happens at least one successful trial for it. Thus, the false key probability for a single candidate is

$$P_F = 1 - (1 - P_2)^{N_c}, \quad (7)$$

where P_2 is computed by Eq. (6). It may seem that P_F is very negligible for the case $\alpha = 0.5$ in which $\gamma = 0.5$. However, this probability can be aggregated over a large number of examined wrong candidates in linear attack, and can lead to an overall false key probability that can not be neglected, as will be seen in simulation results.

B. Analysis of the Designed Attack Strategy for Eve

In [6] Yin et. al. showed that in noisy environment with bit error rate of η , for linear attack on DES cipher, the probability bias of the new linear equation denoted by $\hat{\epsilon}$, as well as the success probability of attacker P_s can be computed based on the linear probability bias of the original linear equation ϵ and the number of obtained pairs by Eve N as

$$P_s = \Phi(2\sqrt{N}\hat{\epsilon} - \Phi^{-1}(1 - 2^{-a-1})), \quad (8)$$

where $\hat{\epsilon} = 2^{u+v}(1 - \eta - 0.5)^{u+v}\epsilon$.

If adversary uses the improved linear analysis technique, she needs to use Matsui's linear equation for DES that requires u bits of plaintext and v bits of corresponding ciphertext where $u + v = 26$ to guess $m = 26$ key bits [22]. As discussed in subsection II-B, in linear attack with bit advantage of a , the total number of examined keys is 2^{56-a} . If the ciphertexts that Eve obtains are error-free, her success probability will be P_s which is the probability that the correct key is among top 2^{56-a} examined candidates. However, when her obtained ciphertexts are erroneous, it is still likely for her to obtain the frame key. Also, it is possible that she gets no frame key either right or wrong for decryption, which imposes her to erase the whole frame. These events have probabilities that are called total success probability and frame erasure probability, respectively, that can be computed based on the following theorem which is proven in appendix A.

Theorem 1: Consider a linear attack with bit advantage of a . Assume Eve's obtained ciphertexts contain bit errors with the rate of η , and that she uses the designed strategy in brute-force step of the linear attack. When Eve examines the right key, she misses it with the probability of P_m given in Eq. (4), and when the key is wrong, she may accept it wrongly with the probability of P_F given in Eq. (7). Let P_s , given in Eq. (8), be the success probability when the ciphertexts are error-free. Then, Eve's total success probability can be computed by

$$P_c = \frac{P_s(1 - P_m)}{P_F 2^{56-a}} [1 - (1 - P_F)^{2^{56-a}}] \approx P_s(1 - P_m), \quad (9)$$

On the other hand, frame erasure probability will be

$$P_e = (1 - P_s)(1 - P_F)^{2^{56-a}} + P_s P_m (1 - P_F)^{(2^{56-a}-1)} \approx [1 - 2^{56-a} P_F][1 - (1 - P_m) P_s]. \quad (10)$$

In addition, the probability that Eve accepts a wrong key in linear attack which we call wrong key probability denoted by P_w can be derived as $P_w = 1 - P_c - P_e$.

Conclusively, we showed that there is possibility that Eve is not able to obtain any key, or to falsely accept a wrong key.

C. Parameter Optimization of Adversary's Attack Strategy

Eave's objective is to mount a successful attack, and in order to achieve this goal, she maximizes the success probability of the utilized linear attack P_c , given in (9), knowing that her computational ability is restricted, and there is a constraint on the number of plaintext/ciphertext pairs that she can accumulate. Namely, she can not perform more than θ DES encryptions. In the linear cryptanalysis designed for noisy environment, the number of all examined keys is 2^{56-a} and each one has to be checked for N_c times. Hence, in the worst scenario Eve has to run $N_c 2^{56-a}$ DES encryptions, which due to Eve's computational restrictions, should not exceed θ . Moreover, we assume that before mounting attack on a frame of data, Eve has already gathered as many number of pairs as data storage capability and time limit allow her denoted by N_{max} . As a result, Eve needs to design attack parameters including N_c , τ and a , to maximize the overall success probability subject to the following constraint

$$\max_{N_c, \tau, a} P_c \quad \text{subject to} \quad \theta \geq N_c \cdot 2^{56-a}, \quad N \leq N_{max}. \quad (11)$$

TABLE II
PARAMETER OPTIMIZATION ALGORITHM FOR ATTACK STRATEGY:

- 1- Initialization: put $\tau = 1$, $N_c = 1$.
Determine T_m and T_f as thresholds for P_m and P_{fault} also N_{cmax} as the maximum value for N_c .
- 2- $\tau \leftarrow \tau + 1$ until $P_{fault} > T_f$ and $\tau < 64$
if $P_{fault} \leq T_f$ or $\tau = 64$ go to the next step
- 3- $N_c \leftarrow N_c + 1$ until $P_m > T_m$ and $N_c < N_{cmax}$
if $P_m \leq T_m$ or $N_c = N_{cmax}$ go to the next step
- 4- Compute $a_0 = \lceil 56 - \log_2(\frac{\theta}{N_c}) \rceil$
- 5- Compute P_c for $a_0 \leq a \leq 56$
choose a for which P_c has its largest value.
- 6- Output τ , N_c and a as attack parameters.

From Eq. (9) it can be concluded that P_c falls as P_m increases. Since according to Eq. (4), P_m mainly depends on N_c , we can define threshold T_m for it and find the minimum number of trials N_c for which P_m remains below T_m . According to equations (6) and (7), to decrease P_F we need to reduce τ as much as possible. If we define a threshold T_f for P_{fault} , the minimum possible value for τ according to our discussion in IV-A is the smallest τ for which P_{fault} remains below T_f . Furthermore, Eve has to choose an optimized value for a to have P_c maximized. The algorithm in Table II, is designed to optimize the linear attack parameters to let Eve achieve the maximum success rate P_c , for a given η subject to her restrictions. In this algorithm, P_{fault} and P_m can be computed using equations (2), (4), respectively.

V. MAIN AND WIRE-TAP CHANNEL MODELLING

In this section, we model main and wiretap channels in block level (with 64-bit input and 64-bit output), using a

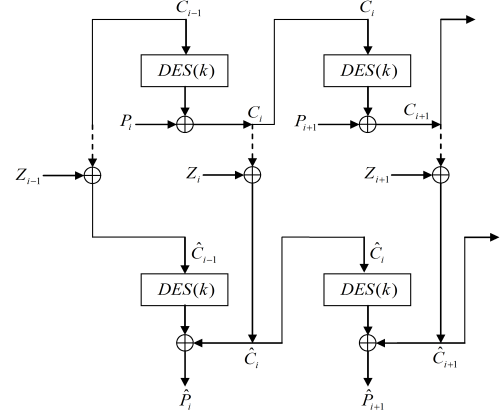


Fig. 4. CFB enciphering and deciphering with channel error

stationary finite state Markov chain (MC). Since Eve might achieve the right frame key, get a wrong one or even get nothing and drop the whole frame, we also need to model her channel in frame level as a three state memoryless channel.

A. Main Channel Modelling Using MC

As it was described, the encrypted data goes through a BSC channel with cross over probability of η , created by intentionally introduced noise in application layer. We next model the CFB cipher, channel with deliberate noise and decipher altogether as a single channel, in order to analyze the effect of intentional noise at the output of decipher. Note that we assume there is no degradation in actual physical channel.

Fig. 4 illustrates the encryption and decryption structure of CFB mode with DES cipher in the presence of introduced noise to ciphertexts. As shown in this figure, $\{C_i\}$ and $\{\hat{C}_i\}$ are the sequences of transmitted 64-bit ciphertext and received noisy ciphertext blocks, respectively, and $\{\hat{P}_i\}$ is the sequence of decrypted blocks at time i for $i = 1, 2, \dots$. In addition, $\{Z_i\}$ is the sequence of 64-bit blocks of intentional bit errors in channel Z_i^j that are independent and identically distributed with Bernoulli distribution as $Pr[Z_i^j = 1] = \eta$ for $j = 1, \dots, 64$, such that $\hat{C}_i = C_i \oplus Z_i$. As Fig. 4 indicates when \hat{C}_i is noisy, it introduces errors with the rate of η to the decryption output at time i , i.e. \hat{P}_i . Moreover, since \hat{C}_{i-1} gets encrypted with DES at time i , due to the avalanche effect, it induces bit error rate of α in \hat{P}_i . As a result, to characterize the channel error state in decryption output at time i , it is required to consider errors in both currently received ciphertext \hat{C}_i and the previous one \hat{C}_{i-1} . Hence, we need to define four states.

Note that in a particular case when we consider $\alpha = 0.5$, we still need to define four states. In this case, when \hat{C}_{i-1} has error, due to the fact that half of the ciphertext will be in error, errors in \hat{P}_i will be independent from \hat{C}_i and consequently from the error state at time $i + 1$. However, when it has no error, errors in \hat{C}_i will affect both decryption outputs at times i and $i + 1$, and therefore the current state will depend on the previous one. As a result, we have to take all four states into account, each with a different transition probability from the input plaintext block P_i denoted as 64-bit vector X to the

output stored plaintext \hat{P}_i denoted by 64-bit vector Y , and let $E = X \oplus Y$ denotes the transition error vector.

The channel states are defined as: state S_0 , in which there is no error from vector X to the vector Y and happens when there is no error in \hat{C}_i and \hat{C}_{i-1} . State S_1 , which happens when there is at least one bit error in \hat{C}_i , but no error in DES cipher input, \hat{C}_{i-1} . State S_2 , which shows the situation in which there is at least one bit error in \hat{C}_{i-1} without any error in \hat{C}_i . In this channel state, due to the avalanche effect, each bit at the output of DES cipher, flips independently with the probability of α causing bit error probability of α in Y . State S_3 , in which both \hat{C}_i and \hat{C}_{i-1} have at least one bit error.

For state S_0 we have $Pr[e_j = 1|S_0] = 0$ and for S_2 , $Pr[e_j = 1|S_2] = \alpha$, where e_j denotes the j^{th} bit of E for $j = 1, \dots, 64$. On the other hand, we should note that in states S_1 and S_3 , output bits can not be treated independently because S_1 and S_3 are based on a given condition on the whole 64-bit ciphertext \hat{C}_i . Let q denote the probability that there exists at least one bit error in Z_i as

$$q = 1 - (1 - \eta)^{64}. \quad (12)$$

The next lemma gives the input-output transition probability for states S_1 and S_3 , which is proven in Appendix B.

Lemma 1: Let X be the input plaintext vector to the CFB encryption mode and Y be the corresponding output of the decryption. If the generated ciphertexts go through a channel with cross over probability of η , we denote the Hamming weight of the resulted error vector E with $W(E)$. Then, for state S_1 the input-output vector transition probability will be

$$Pr(Y|X, S_1) = \begin{cases} \frac{\eta^{W(E)}(1-\eta)^{64-W(E)}}{q} & W(E) \neq 0 \\ 0 & W(E) = 0 \end{cases} \quad (13)$$

where α is the avalanche bit error rate, and γ is given in Eq. (5). The transition probability in state S_3 for all $W(E)$ is

$$Pr(Y|X, S_3) = \frac{\gamma^{W(E)}(1-\gamma)^{64-W(E)} - \alpha^{W(E)}(1-\alpha)^{64-W(E)}(1-q)}{q}. \quad (14)$$

Next, we need to find state transition probabilities. For instance, when the state at time $i-1$ was S_2 , apparently \hat{C}_{i-1} has been error free, so the only condition required to have state S_0 happen at time i is to receive error free \hat{C}_i which has the probability of $1-q$ that is the transition probability from state S_2 to S_0 . Similarly, we can compute other state transition probabilities. Notably, since probability of occurrence of the current state only depends on the previous state, Bob's channel can be modeled as a four state MC that is depicted in Fig. 5 with the following state transition probability matrix:

$$T = \begin{bmatrix} 1-q & q & 0 & 0 \\ 0 & 0 & 1-q & q \\ 1-q & q & 0 & 0 \\ 0 & 0 & 1-q & q \end{bmatrix},$$

whose elements demonstrate the transition probabilities between different states. Note that in each state, input plaintexts undergo different channel conditions and error probabilities. In fact, the main channel can only be modeled as a BSC channel

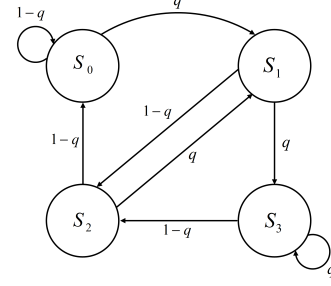


Fig. 5. Alice-Bob channel model as a four state MC

in states S_0 and S_2 with cross over probabilities of 0 and α respectively, whereas in other two states it can be modeled based on input-output transition probabilities in (13) and (14).

In particular, since in MC model for Alice-Bob channel, all four states can be reached from one another, it is an irreducible MC with positive recurrent states [23]. Then, with a supposedly large frame size, MC can reach its stable condition. Since all states are positive recurrent, the set of equations $\mathbf{P}^t \mathbf{T} = \mathbf{P}^t$, and $\mathbf{P}^t \mathbf{1} = \mathbf{1}$ have a unique solution as $\mathbf{P}^t = [p_0, \dots, p_3]$ where p_k denotes the steady state probability of state S_k for $k \in \{0, 1, 2, 3\}$ [23]. Where $\mathbf{1}$ is a 4×1 vector with all elements to be one, and \mathbf{P} is steady state probability vector (SSPV). By solving this equation set, we get

$$\mathbf{P}^t = [(1-q)^2 \quad q(1-q) \quad q(1-q) \quad q^2]. \quad (15)$$

B. Wire-tap Channel Modelling

In section IV-B we showed that adversary can obtain the right key of a frame with the probability of P_c by using optimized verification strategy in linear attack. To consider the worst possible case, we assume that before starting the attack, Eve has gathered the required number of pairs such that for each frame, she has already mounted her attack. When she has been able to achieve the correct key, there will not be any difference between the main channel and her channel, so her decrypted data in that frame undergoes the same channel condition as Bob's. As shown in Fig. 6, we refer to this channel state for Eve as the correct key state in frame level which occurs with the probability of P_c and can be modeled as a MC with four channel states in block level.

Nevertheless, with the probability of P_e , Eve will not be able to get any key for the attacked frame and has to drop the whole frame. We refer to this state as erasure state. Moreover, Eve gets a wrong key with the probability of P_w , such that after using a wrong key due to the avalanche effect in DES cipher, each bit in DES output will be independently flipped with the probability of α . This induced error Xors with intentional i.i.d. channel noise that has bit error probability of η . Consequently, in wrong key state, Eve's channel can be modeled as a BSC with cross over probability of γ given in (5). Conclusively, wiretap channel is a degraded version of the main channel that only in the correct key state can it be as good as Bob's channel. In fact, Eve's channel behaves like a pseudo two-dimensional Markov Chain (P2DMC) [24] with three memoryless states in frame dimension, each acting like another MC in block dimension as shown in Fig. 6.

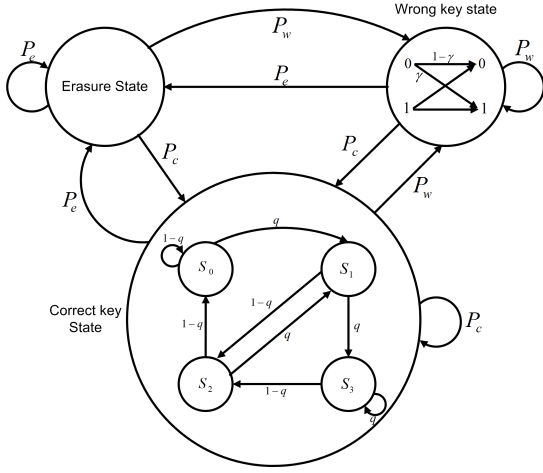


Fig. 6. Eve's hierarchical channel model

VI. SECRECY CAPACITY COMPUTATION

The next step is to quantify the secrecy capacity of the analyzed security system. The capacity of finite state Markov chains was calculated in [25] and [26]. In [27], [28] and [29] the capacity of the finite state Markov chains with binary symmetric channels associated in each state, was studied. In [30] secrecy capacity of a wiretap channel modeled as a finite state MC is computed. We assume that the channel states are perfectly known to Bob and Eve, so what we compute is mutual information between the input X and output Y given the channel state, i.e. $I(X; Y|S_l)$. Since all four states of the main channel are in block level, in order to make Bob aware of the channel states in each block, Alice can use an error detection procedure and embed it in each block. For Eve, we assume that she is aware of this error detection procedure which allows her to beware of channel states in block level. In frame level, it is assumed that she knows the correctness state of each used frame key towards the end of each frame. Specially, this can be considered as the best scenario for Eve, providing us a lower bound for secrecy rate.

The main purpose of secrecy capacity computation is to design a secrecy encoder which is applied ahead of the encryption in application layer over multiple frames. Namely, when the message is transmitted at a rate below the secrecy rate to Bob using a Wyner-type encoding technique[31], [5], we can have an arbitrarily small error probability for Bob as well as the maximum entropy for Eve. In the asymptotic sense, by secrecy encoding, users utilize Eve's failures which cause her channel to be a degraded channel compared to Bob's.

A. Capacity of the Main Channel

When channel state information is available, the capacity is the average of capacities that each one of these MC states contribute to the overall channel capacity [25], [27]:

$$C = \sum_{k=0}^{K-1} p_k C(S_k), \quad (16)$$

where $C(S_k)$ is the channel capacity in state S_k in bit per channel use. It can be computed as the maximum information

rate between input and output vectors, X and Y , respectively, assuming that the current state S_k is known to Bob:

$$C(S_k) = \max_{P_X} I(X; Y|S_k)/64. \quad (17)$$

Note that our modeled four state Markov channel is uniformly symmetric because in any state, channel is output symmetric [25]. For instance, in states S_0 and S_2 , the channel behaves as a BSC channel. In states S_1 and S_3 , if we define the transition probability matrix as $P_{ij} = Pr(Y = j|X = i, S_l)$ for $i \in \mathcal{Y}, i \in \mathcal{X}, l = 1, 3$, its rows and columns are permutations of each other because according to equations (13) and (14), its elements only depend on the HW difference of input-output vectors. As a result, also in states S_1 and S_3 , the channel is output symmetric. In [25] it is shown that for uniformly symmetric channel in which noise is independent of inputs, like our modeled Markov channel, capacity can be achieved with distribution which is uniform and iid. Accordingly, in this finite state Markov channel by uniformly distributed inputs, the mutual information will be essentially maximized.

In state S_0 , channel is an error-free BSC with capacity of 1, i.e. $C(S_1) = 1$, and in state S_2 , it acts like a BSC with cross over probability of α and the capacity of $C(S_2) = 1 - h(\alpha)$, where h is binary entropy function. However, for S_1 and S_3 in which decryption bit errors are not independent, we need to compute the mutual information between input and output vectors, namely $I(X; Y|S_l)$ for $l = 1, 3$, that is

$$I(X; Y|S_l) = H(Y|S_l) - H(Y|X, S_l). \quad (18)$$

We assume that channel state is perfectly known to Bob. In the following theorem which is proven (in Appendix C) using Lemma 1, we compute $H(Y|X, S_l)$ for $l = 1, 3$.

Lemma 2: Consider our four state MC model for the main channel with input vector X and output vector Y . with equally likely input plaintexts, we can compute $H(Y|X, S_1)$ as

$$H(Y|X, S_1) = \frac{-1}{q} \sum_{k=1}^{64} \binom{64}{k} \eta^k (1 - \eta)^{64-k} \cdot \log \left[\frac{\eta^k (1 - \eta)^{64-k}}{q} \right]. \quad (19)$$

and $H(Y|X, S_3)$ will be

$$H(Y|X, S_3) = \frac{-1}{q} \sum_{k=0}^{64} \binom{64}{k} \cdot [\gamma^k (1 - \gamma)^{64-k} - \alpha^k (1 - \alpha)^{64-k} (1 - q)] \cdot \log \left[\frac{\gamma^k (1 - \gamma)^{64-k} - \alpha^k (1 - \alpha)^{64-k} (1 - q)}{q} \right]. \quad (20)$$

On the other hand, for both states S_1 and S_3 , every output vector Y_j can be generated by introducing all possible error vectors over their corresponding input vectors. Hence, since all 64-bit input plaintexts are uniformly distributed, the output will also be equally likely and uniformly distributed. Hence, for $l = 1, 3$ the output entropy is $H(Y|S_l) = 64$. Thus, by using Eq. (18) we can compute the mutual information for states S_1 and S_3 as

$$I(X; Y|S_l) = 64 - H(Y|X, S_l), \quad \text{for } l = 1, 3, \quad (21)$$

where $H(Y|S_1, X)$ is given in Eq. (19), and $H(Y|S_3, X)$ in Eq. (20). According to Eq. (17) the channel capacity in states S_l for $l = 1, 3$ will be

$$C(S_l) = \frac{I(X; Y|S_l)}{64} \quad (\text{bits per channel use}). \quad (22)$$

where $I(X; Y|S_1)$ and $I(X; Y|S_3)$ are given in Eq. (21). We can analyze Alice-Bob channel as a finite state MC with steady state probabilities given in Eq. (15). Hence, according to Eq. (16) Bob's channel capacity C_B as the average of the state capacities can be computed as

$$C_B = (1 - q)^2 + q(1 - q)[C(S_1) + 1 - h(\alpha)] + q^2 C(S_3). \quad (23)$$

where α is the average bit error rate caused by the avalanche effect. In addition, $C(S_1)$ and $C(S_3)$ are given in Eq. (22), implying that these capacities mainly depend on q , γ , α and η . As a result, the main channel capacity depends on q and γ which according to Eq.'s (12) and (5) are themselves functions of η , for a fixed α . Therefore, Bob's channel capacity mainly depends on the original channel cross over probability η .

B. Secrecy Capacity of the Wire-tap Channel with Noise

As discussed in subsection V-B, wiretap channel is a degraded version of the main channel that only in correct key state can be as good as Bob's channel. In the worst possible scenario, we assume she is perfectly aware of channel states. When Eve with the probability of P_c obtains the right key, her channel capacity will be the same as Bob's, i.e. C_B , but when with the probability of P_w gets a wrong key, her channel will turn into a BSC with the cross over probability of γ , which has the capacity of $1 - h(\gamma)$. Note that, the erasure state does not contribute to the capacity. Hence, Eve's capacity will be

$$C_E = P_w(1 - h(\gamma)) + P_c C_B, \quad (24)$$

where C_B is given in Eq. (23). In the following theorem secrecy capacity is found whose proof is given in Appendix D.

Theorem 2: The secrecy capacity for the created wire-tap channel with the described channel models for Bob and Eve will be

$$C_s = C_B(1 - P_c) - (1 - P_e - P_c)(1 - h(\gamma)). \quad (25)$$

This result implies that secrecy capacity mainly depends on P_c , P_e and C_B . Due to the fact that all P_c , P_e and C_B highly depend on the channel error rate η , the main parameter that impacts secrecy capacity of the system is intentional noise. Namely, if Alice can control the cross over probability of the channel, it is possible to adjust secrecy rate of the system. Note that Alice applies secrecy encoding over multiple frames in order to statistically average over Eve's possible failures in frame level, and also to enable Bob to do the error correction coding when burst of errors occurs. Basically, Alice and Bob has to use a well designed wiretap channel encoder, based on the computed secrecy rate in Eq. (25). Notably, the main issue in this scheme is delay that is imposed on the system by applying multiple frame encoding that makes this scheme applicable only for delay tolerant communication.

VII. NUMERICAL RESULTS

The main objective of numerical analysis is to evaluate the effect of varying η on secrecy rate in order to see if there exists an optimum value for η for which secrecy capacity reaches its maximum. In simulations, we assume that Alice by controlling η is able to generate a degraded wiretap channel. In addition, we assume that the whole data is divided into equal size frames, each containing as many number of 64-bit data blocks as four-state MC reaches its steady state, such that for each frame, encryption and decryption key remains constant.

Let us assume that $\theta = 2^{48}$ is the maximum number of DES encryptions that Eve can perform to establish an attack on each frame. Because for instance, with a CPU having speed of 2.6 GHz, it takes for about 30 hours for her to accomplish these many encryptions. For attack optimization algorithm proposed in section IV-C, the initial values selected for n is $n_0 = 20$, maximum possible value for N_c is chosen $N_{cmax} = 100$, and the thresholds T_f and T_m are set to 10^{-5} . Furthermore, we chose α as avalanche effect bit error rate to be 0.5. To evaluate the effect of noise variation on the performance of the system, we changed η from 10^{-4} to 0.05 with 500 steps of size 10^{-4} . Moreover, suppose that Eve is able to detect these step size changes on η by probing the channel and each time is able to optimize all attack parameters using the parameter optimization algorithm. We assume that Eve is not allowed to use more than $N_{max} = 2^{46}$ number of pairs, and prior to attack on each frame, she obtains the required number of plaintext/ciphertext pairs and mounts her attack.

In Fig. 7, overall success probability, wrong key and frame erasure probabilities are depicted as functions of η for fixed number of pairs equal to 2^{46} . As this Figure displays with rising η , P_c is monotonically decreasing, reaching zero for $\eta > 0.017$, while wrong key probability P_w goes to 1 for $\eta = 0.05$ because of increase in P_F . As discussed in section IV-A, the obtained results for P_w show that it becomes considerable for some channel conditions and can not be ignored. In Fig. 8 curves of main and wiretap channel capacities as well as the secrecy capacity are drawn as functions of η . This Figure shows that Alice-Bob channel capacity is monotonically decreasing with increase in η while secrecy capacity C_s rises up to its maximum value 0.3442 for $\eta = 0.0125$ and then falls. Indeed, this cross over probability can be considered optimum value for which secrecy capacity achieves its maximum.

TABLE III
OPTIMIZED ATTACK PARAMETERS USING PROPOSED ALGORITHM IN IV-C

η	0.001	0.005	0.01	0.0125
N_c	5	9	16	20
τ	3	5	6	7
a	23	24	24	27
P_c	0.9999	0.9636	0.5014	0.1618

In Table III optimized attack parameters using our proposed algorithm for four different η 's, i.e. 0.001, 0.005, 0.01 and 0.0125 are given. According to this table, with increase in η , the required number of trials N_c for each key increases from 5 to 20 in order to keep P_m below the threshold $T_m = 10^{-5}$ when it rises. The same holds for parameters a and τ which to

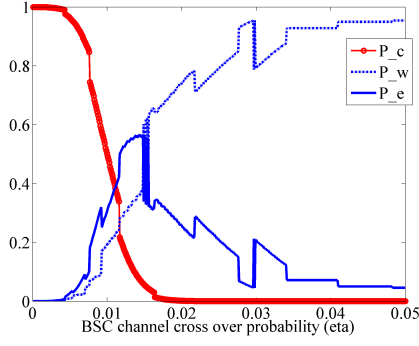


Fig. 7. Overall success probability, frame erasure and wrong key probabilities versus channel cross over probability

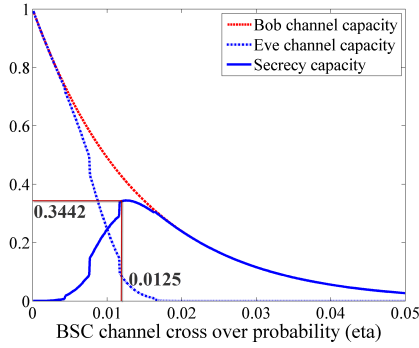


Fig. 8. Main channel and Eve's channel capacities and secrecy capacity for varying channel cross over probability

achieve the determined thresholds, have to increase with rising channel noise to maximize the overall success probability. According to our numerical results, Alice can adjust channel conditions by introducing deliberate noise in application layer to have $\eta = 0.0125$, to achieve the desirable secrecy capacity.

VIII. CONCLUSION

In this paper we showed that by introducing tunable noise in application layer upon the encrypted data, even though Eve utilizes an optimized attack strategy, the secrecy rate of the system can remarkably increase. In fact, Alice can achieve a sufficiently large secrecy capacity by adjusting the cross over probability of the channel using deliberate noise. This secrecy rate guarantees a highly secure and reliable communication using wiretap channel coding in application layer over multiple frames. For secrecy capacity computation we tailored the known channel states scenario. In our future work, we will focus on the unknown state case and also will consider a more generic cipher. In addition, we will work on more detailed design of a secrecy encoder in this framework.

APPENDIX A

PROOF OF THEOREM 1

Proof: Suppose that all possible 2^{56-a} key candidates are arranged as $k_1, k_2, \dots, k_{2^{56-a}}$ from the lowest rank to the highest. Let H_i be the hypothesis that k_i is the original key and $g_i = 1$ be the event that Eve decides that k_i is correct.

We define a Bernoulli random variable B which is equal to 1 when the right key is among top 2^{56-a} candidates, and 0, otherwise. Thus, $Pr[B = 0] = 1 - P_s$ and $Pr[B = 1] = P_s$. Let P_c be the total success probability for Eve. Note that when $B = 0$, the right key will not be tested and consequently can not be found. Therefore, we have

$$P_c = \sum_{i=1}^{2^{56-a}} Pr[g_i = 1, H_i | B = 1] \cdot Pr[B = 1]. \quad (26)$$

The probability that Eve can realize the right key k_i is

$$Pr[g_i = 1, H_i | B = 1] = Pr[g_i = 1 | H_i, B = 1] \cdot Pr[H_i | B = 1].$$

For Eve to be able to find the correct key at rank i , since she starts the test from upper ranks to the lower ones, there should not be any false key acceptance for ranks higher than i , as well as a key missing event for rank i . Hence,

$$Pr[g_i = 1 | H_i, B = 1] = (1 - P_F)^{2^{56-a}-i} (1 - P_m). \quad (27)$$

Moreover, Decisions about all 2^{56-a} keys are independent, and all of the tested keys are equally probable to be the right one, i.e. $Pr[H_i | B = 1] = \frac{1}{2^{56-a}}$. Therefore, by using Eq.'s (26) and (27), we obtain Eq. (9) for total success probability.

The next step is to compute the frame erasure probability. Assume that the right key is k_i and is located among top 2^{56-a} candidates. In order to obtain no key, Eve should not have any false key admission for k_j , $j \neq i$ for $i, j = 2^{56} - 2^{56-a} + 1, \dots, 2^{56}$, i.e. top 2^{56-a} candidates except the right key itself, and in addition to that she has to miss the right key k_i . When k_i is not among top candidates, since it will not be examined, Eve gets nothing provided that there has been no wrong key acceptance event for top 2^{56-a} tested candidates. As a result, frame erasure probability can be computed according to Eq. (10). By a similar technique, we can prove that the wrong key probability is $P_w = 1 - P_e - P_c$. ■

APPENDIX B

PROOF OF LEMMA 1

Proof: We need to compute vector transition probabilities between all possible input and output vectors X and Y for states S_1 and S_3 . Hence, for $k = 1, 3$

$$Pr[Y|X, S_k] = Pr[X \oplus Y|X, S_k] = Pr[E|S_k], \quad (28)$$

where E is the decryption error vector which is bit-wise Xor of input and output vectors. The last equality is because E depends on channel errors in previous and current ciphertexts, so given the state, it is independent from input vector X . To analyze states S_0 and S_1 , we define two events, A and B as

- A : There exists at least one bit error in \hat{C}_i
- B : There exists at least one bit error in \hat{C}_{i-1} .

As a result, $S_1 = A \cap \bar{B}$, and we can write

$$Pr(E|S_1) = Pr(E|A, \bar{B}) = \frac{Pr(E|\bar{B})Pr(A|E, \bar{B})}{Pr(A|\bar{B})}. \quad (29)$$

The fact that events A and B are caused by two independent channel error vectors Z_{i-1} and Z_i implies that A is independent of B and its complementary, i.e. $Pr(A|\bar{B}) = Pr(A) = q$.

When event B has not occurred, since only \hat{C}_i can induce bit errors with rate of η into the stored plaintext, the probability that a particular decryption error vector E with Hamming weight of $W(E)$ takes place will be

$$Pr(E|\bar{B}) = \eta^{W(E)}(1 - \eta)^{64-W(E)}. \quad (30)$$

In state S_1 , HW of error vector E can not be zero because we know that the only source that can induce error at stage i is Z_i that surely has a non-zero bit. In this case, given an error vector E with $W(E) \neq 0$ and knowing that event B did not occur, we can infer that this error is induced by error in \hat{C}_i , hence event A has certainly occurred, i.e. $Pr(A|E, \bar{B}) = 1$. Thus, using equations (28), (29) and (30), we can obtain the input-output transition probability in S_1 as in Eq. (13).

On the other hand, according to its definition, state S_3 takes place when both events A and B happen, i.e. $S_3 = A \cap B$. Therefore,

$$Pr(E|S_3) = Pr(E|A, B) = \frac{Pr(E|B)Pr(A|E, B)}{Pr(A|B)}. \quad (31)$$

Knowing that B occurred, implies that there exists one bit error in DES input, which induces independent bit errors with the rate of α in cipher output and consequently in \hat{P}_i , but also there is independent bit error sequence caused by \hat{C}_i that has the rate of η . Since the decryption error vector E_i is a result of Xor of these two error sequences, we can say that E_i is a sequence of random bits with i.i.d. distribution and bit error probability of γ which is given in Eq. (5). As a result,

$$Pr(E|B) = \gamma^{W(E)}(1 - \gamma)^{64-W(E)}, \quad (32)$$

so we can write

$$Pr(A|E, B) = 1 - Pr(\bar{A}|E, B) = 1 - \frac{Pr(E|\bar{A}, B)Pr(\bar{A}|B)}{Pr(E|B)}. \quad (33)$$

When A has not occurred, But B has, the only error source will be the cipher input that induces independent bit errors with the rate of α into the output. Consequently, we have

$$Pr(E|\bar{A}, B) = \alpha^{W(E)}(1 - \alpha)^{64-W(E)}. \quad (34)$$

Then, using equation (32), (33) and (34) gives us

$$Pr(A|E, B) = 1 - \frac{\alpha^{W(E)}(1 - \alpha)^{64-W(E)}(1 - q)}{\gamma^{W(E)}(1 - \gamma)^{64-W(E)}}. \quad (35)$$

Finally, according to equations (28), (31) and (35) the input-output transition probability in state S_3 for all $W(E)$ can be obtained using Eq. (14). ■

APPENDIX C PROOF OF LEMMA 2

Proof: If we assume that all 2^{64} possible input plaintexts are equally likely, for $l = 1, 3$ we can write

$$\begin{aligned} H(Y|S_l, X) &= - \sum_i \sum_j \frac{1}{2^{64}} Pr(Y = Y_j | X = X_i, S_l) \\ &\quad \cdot \log Pr(Y = Y_j | X = X_i, S_l) \\ &= - \sum_{j=1}^{2^{64}} Pr(E = E_j | S_l) \log Pr(E = E_j | S_l). \end{aligned} \quad (36)$$

The second equality is resulted from Eq. (28) for $E_{i,j} = X_i \oplus Y_j$ as the decryption error vector. Furthermore, for state S_1 as discussed in subsection V-A, Hamming weight of the error vector E can not be zero. Thus, we can take E_1 as a 64-bit zero vector and exclude it from this summation. Then, using Eq. (13) brings about the following result

$$\begin{aligned} H(Y|S_1, X) &= \frac{-1}{q} \sum_{j=2}^{2^{64}} \eta^{W(E_j)}(1 - \eta)^{64-W(E_j)} \\ &\quad \cdot \log \left[\frac{\eta^{W(E_j)}(1 - \eta)^{64-W(E_j)}}{q} \right]. \end{aligned} \quad (37)$$

We know that out of all 2^{64} error vectors, the number of possible vectors with Hamming weight of W or vectors with W non-zero bits is the number of possibilities of choosing W bits out of 64 bits which is equal to W -combinations from 64 elements. Finally, Eq. (37) can be rewritten as Eq. (19). Note that we excluded zero weight case, i.e. $k = 0$.

For state S_3 , we can compute $H(Y|S_3, X)$ using Eq. (36) for $l = 3$. In this case, $j = 1$ is not excluded because unlike state S_1 in state S_3 , it is possible to have decryption error vector E_1 with zero weight. Finally, by using Eq. (14) similar to the entropy in S_1 , we obtain $H(Y|S_3, X)$ in Eq. (20). ■

APPENDIX D PROOF OF THEOREM 2

Proof: It is shown in [32] that when the mutual information between Alice at Bob and the mutual information between Alice and Eve are individually maximized by the the same input distribution, and the main channel is less noisy that the wiretap channel, the secrecy capacity can be computed as the difference of two capacities. In our channel model, since both Bob and Eve's mutual information with Alice are maximized with uniformly distributed inputs X , and wiretap channel is noisier that the main channel, the secrecy capacity will be, $C_s = C_B - C_E$. It gives us the final result in Eq. (25). ■

REFERENCES

- [1] W. F. Ehrsam, S. M. Matyas, C. H. Meyer, and W. L. Tuchman, "A cryptographic key management scheme for implementing the data encryption standard," *IBM Systems Journal*, vol. 17, no. 2, pp. 106–125, 1978.
- [2] B. Schneier, *Applied cryptography (2nd ed.): protocols, algorithms, and source code in C*. New York, NY, USA: John Wiley & Sons, Inc., 1995.
- [3] A. Wyner, "The wire-tap channel," *Bell Syst. Tech. J.*, vol. 54, pp. 1355–1387, 1975.
- [4] I. Csiszar and J. Korner, "Broadcast channels with confidential messages," *IEEE Trans. Inform. Theory*, p. 339348, May 1978.
- [5] A. Thangaraj, S. Dihidar, A. Calderbank, S. McLaughlin, and J. Merolla, "Applications of LDPC codes to the wiretap channel," *IEEE Transactions on Information Theory*, vol. 53, no. 8, pp. 2933–2945, Aug 2007.
- [6] R. Yin, S. Wei, J. Yuan, X. Shan, and X. Wang, "Tradeoff between reliability and security in block ciphering systems with physical channel errors," *Proc. IEEE Military Commun. Conf. (MILCOM)*, 2010.
- [7] S. Goel and R. Negi, "Guaranteeing secrecy using artificial noise," *Wireless Communications, IEEE Transactions on*, vol. 7, no. 6, pp. 2180–2189, june 2008.
- [8] J. Vilela, M. Bloch, J. Barros, and S. McLaughlin, "Wireless security regions with friendly jamming," *Information Forensics and Security, IEEE Transactions on*, vol. 6, no. 2, pp. 256–266, june 2011.
- [9] M. J. Mihaljevic and H. Imai, "An approach for stream ciphers design based on joint computing over random and secret data," *Computing*, vol. 85, pp. 153–168, 2009.

- [10] M. Willett, "Deliberate noise in a modern cryptographic system (corresp.)," *IEEE Transactions on Information Theory*, vol. 26, no. 1, pp. 102–104, 1980.
- [11] M. Mihaljevic and F. Oggier, "A wire-tap approach to enhance security in communication systems using the encoding-encryption paradigm," *IEEE 17th International Conference on Telecommunications (ICT)*, pp. 83–88, April 2010.
- [12] P. C. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO '99. London, UK: Springer-Verlag, 1999, pp. 388–397.
- [13] T. Roche, V. Lomn, and K. Khalfallah, "Combined fault and side-channel attack on protected implementations of AES," *CARDIS*, pp. 65–83, 2011.
- [14] Y. Liu, P. Chen, G. Xie, Z. Liu, and Z. Li, "The design of a low-power asynchronous DES coprocessor for sensor network encryption," in *International Symposium on Computer Science and Computational Technology (ISCST)*, vol. 2, Dec 2008, pp. 190–193.
- [15] W. Zibideh and M. Matalgah, "Modified-DES encryption algorithm with improved BER performance in wireless communication," in *Radio and Wireless Symposium (RWS), 2011 IEEE*, jan. 2011, pp. 219–222.
- [16] Y. Xiao, H. Chen, X. Du, and M. Guizani, "Stream-based cipher feedback mode in wireless error channel," *IEEE Trans. Wireless Comm.*, vol. 8, pp. 622–626, 2009.
- [17] H. Heys and S. Tavares, "Avalanche characteristics of substitution-permutation encryption networks," *IEEE Trans. Comput.*, vol. 44, no. 9, pp. 1131–1139, Sep 1995.
- [18] K. Nyberg, "S-boxes and round functions with controllable linearity and differential uniformity," in *FSE*, pp. 111–130, 1994.
- [19] M. Matsui, "Linear cryptanalysis method for DES cipher," *Lecture Notes in Computer Science*, vol. 765, pp. 385–397, 1994.
- [20] A. Selcuk and A. Bicak, "On probability of success in linear and differential cryptanalysis," *SCN 2002*, pp. 174–185, 2003.
- [21] C. H. Bennett, G. Brassard, C. Crepeau, and U. M. Maurer, "Generalized privacy amplification," *IEEE Trans. Inform. Theory*, vol. 41, pp. 1915–1923, Nov. 1995.
- [22] M. Matsui, "The first experimental cryptanalysis of the data encryption standard," *Lecture Notes in Computer Science*, vol. 835, pp. 1–11, 1994.
- [23] S. Ross, "Introduction to probability models," *University of Southern California, Academic Press*, Tenth Edition, ISBN: 978-0-12-375686-2, 2010.
- [24] S. Yu, Z. Liu, M. Squillante, C. Xia, and L. Zhang, "A hidden semi-markov model for web workload self-similarity," *21st IEEE International Performance, Computing, and Communications Conference*, pp. 65–72, 2002.
- [25] A. J. Goldsmith and P. P. Varaiya, "Capacity, mutual information, and coding for finite-state markov channels," *IEEE Trans. Inform. Theory*, vol. 43, pp. 868–886, May 1996.
- [26] T. Holliday, A. Goldsmith, and P. Glynn, "Capacity of finite state markov channels with general inputs," in *Proceedings of the IEEE International Symposium on Information Theory* 289, 2003.
- [27] H. S. Wang and N. Moayeri, "Finite-state markov channel: A useful model for radio communication channel," *Proc. IEEE Veh. Tech. Conf. (VTC)*, vol. 44, pp. 163–171, Feb 1995.
- [28] M. Mushkin and I. Bar-David, "Capacity and coding for the Gilbert Elliot channel," *IEEE Trans. Inform. Theory*, vol. 35, pp. 1277–1290, 1989.
- [29] A. Lapidot and I. E. Telatar, "The compound channel capacity of a class of finite-state channels," *IEEE Trans. Inform. Theory*, vol. 44, pp. 973–983, 1998.
- [30] Y. Sankarasubramaniam, A. Thangaraj, and K. Viswanathan, "Finite-state wiretap channels: Secrecy under memory constraints," *Information Theory Workshop, 2009. ITW 2009. IEEE*, pp. 115–119, Oct. 2009.
- [31] L. H. Ozarow and A. D. Wyner, "Wire-tap channel II," *Bell System Technical Journal*, vol. 63, no. 10, pp. 2135–2157, Dec 1984.
- [32] M. v. Dijk, "On a special class of broadcast channels with confidential messages," *IEEE Trans. Inform. Theory*, vol. 43, pp. 712–714, Mar 1997.